Module 1

Fundamentals of the World Wide Web

The Goals of this Module

- Understand basic Internet terms
- Gain an overview of how web pages are built using HTML
- Learn basic HTML tags and attributes
- Investigate how different browsers interpret web pages
- Understand how Dreamweaver 4 generates HTML
It seems like the Internet is completely inescapable these days. Everywhere you turn, you see another advertisement for the latest web site or service or another story about someone making (or losing) millions of dollars on their hot Internet idea. For many people, using the Internet has become as common as using the telephone. In fact, entirely new terms have been coined that today seem completely familiar to most people, such as dot coms, surfing the Net, e-business—the list goes on and on. But how many people know what the Internet is really all about, such as how it works, how web pages are accessed, and how sounds and animations get created and placed so that we can get to them? At its heart, this book is all about how those things happen—from designing web pages, to creating and inserting pictures and graphics, to crafting sophisticated animations that move and react to our commands. By following along with the exercises in this book, you'll soon be creating dynamic web pages with original graphics, like the one shown in Figure 1-1. But to do those things, you first need to know how they work.

Understanding the Web—Basic Internet Terminologies

With the Internet seemingly everywhere, it’s hard to believe that the Net as we know it has been around less than ten years. While the basic structure of the Internet was created during the 1960s and 1970s, when scientists who needed a way to share their research created an interconnected network of computers, the modern Internet wasn’t born until the arrival of the first software program that allowed the average consumer to look at pages that included colors and pictures, and to move to another page by clicking a link with their mouse. A software program with those capabilities is known as a web browser.

The first popular graphical web browser was created by a group of graduate students led by Marc Andreessen and Eric Bina at the University of Illinois. Called Mosaic for X, it was first released in 1993. Marc Andreessen went on to start Netscape Corporation, and his revised browser became the cornerstone of our modern online experience. The Netscape browser is still with us, of course, and it’s been joined by browser programs created by Microsoft (Internet Explorer) and by other browsers as well.

All browsers have the same basic functions: to read a set of instructions, or code, that directs our computers to display text and pictures, enable us to get files from another computer (download), or send information to a computer
Module 1: Fundamentals of the World Wide Web

The exercises in this book will soon have you creating your own unique web pages for publication to the Web.

(upload) and receive a response. While that seems pretty simple, in the short history of the Internet, the capabilities of browsers have expanded many times over, and today you not only can see a simple picture displayed at the top of a web page with some text below it, but also can track an airplane in flight, make reservations for a movie at the cinema, or chat with people all over the world. Yet even though the Internet has become incredibly sophisticated, the basic function of every web page you see is controlled by the code that is read by the browsers on our computers.

What Is HTML?
This basic code for the design of Web pages is called Hypertext Markup Language (HTML), and it is the foundation of every web page on the Web. The next section of this module takes a closer look at HTML.
Notice what the code actually does, though. It gives directions to your computer to perform certain actions, such as display text or images, format the page in a particular layout, and insert objects that you can interact with. As a new web designer, this is one of the first lessons that you have to learn. Most of the actual work is done at the user’s (or client’s) computer. Something you’ll hear over and over again is that you need to design your pages for the user’s, or viewer’s, experience, and that includes considering the fact that their computers control how the page is displayed for them.

Your job is to provide pages that neither get bogged down with long download times nor consume so much of the users’ computer resources that they leave your page out of frustration. And, of course, you need pages that attract viewers because the content is something they want to see. Thankfully, the technical challenge of designing quick-loading web pages is much easier to address when working with programs like Dreamweaver and Fireworks, because managing and optimizing your pages and files is the primary function of these programs.

Getting Connected
What happens after you log on to your Internet Service Provider (ISP), the company that provides your connection to the Internet? Your computer takes the first action when the browser that you’re using transmits a request to a remote computer to show you your initial web page. That remote computer—known as a server—is the place where all the files are stored that are necessary for your starting page to display properly. In fact, as you’ll see when HTML is discussed later in the chapter, web pages are not like printed pages at all. That is, the images and other information on the page are not one object, like a page in a magazine. Instead, the page contains code that tells the browser to retrieve and display those images, as shown in Figure 1-2. The browser’s job is to bring the images all together and display them for you, which is why your pages don’t always load all at once, but tend to display the text first and then the images as they are received from the server.

Your home page and every other web page on the Internet have some basic things in common. The page is part of a web site—a collection of web pages, files, and links all associated with a particular domain name. These domain names are purchased by companies or individuals (such as www.amazon.com, www.yahoo.com, and www.msnbc.com), registered to nonprofit groups (such as www.pbs.org and www.splc.org), or assigned to government agencies or educational institutions (www.whitehouse.gov, www.stetson.edu, and www.firm.edu, for example) as a way to identify them as unique locations on
Module 1: Fundamentals of the World Wide Web

the World Wide Web (or simply the Web), the collection of servers that store the files of the web sites. The most common current domain suffixes (the three-letter code after the dot) are provided in Table 1-1.

Seem confusing? It’s really not. Just as you have a unique address for your home, each web site needs a unique address so that it can be found. Depending on your browser, this address may be shown in the Location bar (Netscape) or the Address bar (Internet Explorer). Either way, it means the same thing: it identifies to the server exactly where the files you are requesting can be found. Those addresses are called universal resource locators (URLs).

As you’ll see when you start working with Dreamweaver, web site managers (or webmasters) have a lot to do with keeping the files and all the supporting

<table>
<thead>
<tr>
<th>Domain Suffix</th>
<th>Domain Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>.com</td>
<td>Commercial, for-profit web site</td>
</tr>
<tr>
<td>.org</td>
<td>Nonprofit organization</td>
</tr>
<tr>
<td>.gov</td>
<td>Government agency</td>
</tr>
<tr>
<td>.net</td>
<td>Internet service provider</td>
</tr>
<tr>
<td>.mil</td>
<td>Military</td>
</tr>
<tr>
<td>.edu</td>
<td>Educational institution</td>
</tr>
<tr>
<td>.k12</td>
<td>Kindergarten through high school</td>
</tr>
</tbody>
</table>

Table 1-1 Common Domain Names on the World Wide Web
assets of their web pages organized and in proper working order. As a client of a browser program, all you really care about is getting the page to load quickly and properly and getting to the information you want to see.

Do you remember the first time you used the Internet? You probably were fascinated by how easy it was to move from one page to another, all with the click of a mouse. It’s no accident that the term “web surfing” was coined as a way to describe the experience of moving effortlessly from page to page. What makes all that possible? Again, it’s all controlled by the instructions written into the code. One of the things that makes HTML so valuable is its ability to insert links (or hyperlinks) in each page. In the same way that HTML allows a browser to display images, it can also create instructions to go to another web page or another section of a page when the user clicks their mouse on an image or string of text. Without this ability, web pages would be static, immovable objects—far different from the dynamic and interactive experience of the Web today.

In a nutshell, the Internet is simply (simply!) a huge worldwide interconnected network of computers, all using a common language, that allows users to retrieve information stored on remote computers, and display it on their computers at home, work, or school. This is all done through the magic of a programming language, HTML, that lets web page designers insert images, text, sounds, and other objects into their pages, through the browser programs that read the instructions in the code and display the results on the user’s computer. Hmmm. When stated that way, it doesn’t seem hard at all, does it? And, as you’ll see in the next section, gaining a basic understanding of HTML isn’t all that difficult either!

1-Minute Drill

- What is the basic function of a web browser?
- Define the term web site.

Getting a Handle on HTML

Just saying the words “programming language” to some people can cause their eyes to glaze over and their senses to go numb. Other people are fascinated by the challenges presented by learning to program computers and by sorting out
all the instructions necessary to get a computer to perform the way they want it to. Whether you’re a natural programmer or not, however, you need a basic understanding of HTML to go very far as a web designer. Although Dreamweaver (and Fireworks, as you’ll see) will certainly do the majority of the programming for you, by gaining an understanding of what’s going on under the hood of your web page, you’ll have a better chance of tracking down problems and getting them corrected when something isn’t working just right. You’ll also see as you work with Dreamweaver that you can save yourself a lot of work by going directly to the point on your page where you want to do something. To do that, you need to know about tags.

**HTML Tags**

You'll start by looking at what a page displayed in its raw HTML state looks like. If you haven’t done so yet, download and unzip the files located in module_one.zip at www.osborne.com. After you unzip the files, use your browser to open the file called html_one.htm.

After your browser displays the page, select on View | Page Source (for Netscape), or View Source (for Internet Explorer). You should see something like this:

```html
<html>
<head>
<title></title>
</head>
<body>This is a very simple web page.
</body>
</html>
```

while your browser displays this:

```
This is a very simple web page.
```
So what’s going on here anyway? Why did it take so much code to display one little sentence? And what’s with all of those left and right arrow brackets in the code? Let’s take a look at that string of code and decipher it.

**Hint**

You’ll notice that the filenames throughout the book use the underscore symbol ( _ ) to separate words. Although modern computer operating systems can easily deal with filenames that have spaces in them, not all servers operate correctly when they come across a space. For that reason, you won’t see any spaces in exercise filenames or in files that you’ll create.

The first thing to understand about HTML is that it is written in a series of commands known as *tags*. Tags are the commands written by the programmer (or by Dreamweaver or another web authoring program) that tell the browser what to do. Take a look at the tags in the preceding example:

```html
<html>
<head>
<title></title>
</head>
<body>This is a very simple web page.</body>
</html>
```

Notice first of all that all the tags are written as pairs, just as parentheses are always written in pairs (like this). And, just as you need opening and closing parentheses, an opening tag and a closing tag are almost always necessary.

Every web page begins and ends with the `<html>` tag. This lets the browser know that it must process the page as an HTML or web document instead of, say, a word processor document or an image file. The brackets are there to let the browser know that a tag is enclosed and that certain actions are expected. Finally, notice that the final (closing) `</html>` tag adds a forward slash inside the left arrow bracket, which lets the browser know that it should stop processing, or close, that tag.

Next, you see the `<head>` tag and, enclosed within it, the `<title>` tag. The `<head>` tag tells the browser that the information inside it goes at the top, or head, of the document, and the `<title>` tag is where the information about the title of the web page is displayed. In the first example, your browser calls this an “Untitled Document” because no information is enclosed by the `<title>` tag. Now, open the file called html_two.htm from the exercise files and you’ll see
that a title has been added to the page. Viewing the source of the page reveals
that more information has been added to the HTML in the example page—the
words “A Simple Web Page” enclosed by the <title> tags.

```
<html>
<head>
<title>A Simple Web Page</title>
</head>
<body>This is a very simple web page.
</body>
</html>
```

**HTML Attributes**

Take a look at another example. Open html_three.htm on your browser and
look at its source, shown here:

```
<html>
<head>
<title>A Simple Web Page</title>
</head>
<body bgcolor="#CCFFCC">
<br>
</div align="center">This is a very simple web page.</br>
</body>
</html>
```

Notice that one new tag, the <div> tag, was added. But something else has
been added—another set of instructions inside the left and right bracket arrows
that enclose the tags. Did you notice the difference when you first opened the
practice file? The page background is now pale green and the text has been
centered in the page. If you look at the code, you can probably guess that bg
color="CCFFCC" is the code to make the background color that shade of green.
And if you look closely at the command inside the <div> tag, you see the added
instructions to align="center".

These added instructions are known as **attributes**, and they are used to do
things such as change the color of text; change the alignment of text, paragraphs,
and images; change the background color; or insert an image in the background
of the page. As Wendy Willard describes in her excellent book, *HTML: A Beginner’s
Guide* (Osborne/McGraw-Hill, 2001), you can think of the tags in a web page as
the ice cream in an ice cream sundae, while the attributes are all the toppings.
Tags provide structure and organization to your pages while attributes make
them tastier!
Dreamweaver 4 Fireworks 4 Studio: A Beginner’s Guide

Take a look at another example. Open html_four.htm in your browser. You can see a number of attributes at work here. The text has been changed quite a bit—different colors have been added, sizes have been changed, and bold and italic styles have been added. Also, a background image has been added to the page, and the final result is as you see in Figure 1-3. The source code reveals that what’s really been done is that a number of tags have been changed by inserting attributes within them. The attributes in the source code are underlined so that you can locate them more easily. You’ll quickly notice that while the way that HTML works isn’t really all that complicated, by the time you look at the code for a complex page, you might have trouble seeing the forest for the trees. Don’t worry! Dreamweaver is going to make this all much simpler for you.

```html
<html>
<head>
<title>A Simple Web Page</Title>
</head>
<body background="grayparchment.gif">
<div align="center">
<p><font color="#0000FF" size="7">This</font><font color="#0000FF" size="5">is a <font color="#FF0000">very</font> <b>simple</b> <i>Web</i><font size="7">page</font>.</font> </p>
<p><img src="images/simpleimage.gif" width="299" height="122" > </p>
<P><a href="http://www.osborne.com">Visit Osborne Publishing.</a></p>
</div>
</body>
</html>
```

What do you notice about the way the attributes are written? If you look closely, you’ll see that they all have a few things in common:

- Attributes are always inside the tag (between the opening and closing tags) that they modify
- The command for the attribute (color means we want to change the color, for instance) comes before an equal sign. Every attribute needs that equal sign!
- The value of the attribute comes after the equal sign and contains the information you want applied.

You also see two attributes that control the images displayed in your page. The first is background, and the second is src. One very important note about images: Your code has to tell the browser not only what the image is—its name and size—but also where the image is.
Notice that the background attribute is different from the src attribute. It only lists the name of the file that is used for the background on your page, because the image is in the same folder as the HTML file itself:

```html
<body background="grayparchment.gif">
```

Because no other information is provided, the browser assumes that this image can be found in the same location as the HTML file itself.

---

**Figure 1-3** The results of the HTML code in html_four

**Hint**

A quick note about colors. On the preceding example web page, the word “This” is blue. Why does the attribute for that word list it as “#0000FF”? Couldn’t you just type `blue` instead? Actually, you could, and you’d get the same color. But what if you wanted a grayish-green blue color? To get more-specific colors, you need more-specific instructions, so when that first consumer browser was developed by Netscape, a way had to be developed to display a variety of colors—all on different computers that might have different settings. For that reason, a list of 216 “Web safe” colors was developed, written in a type of code called *hexadecimal*. This code uses letters and numbers to specify exactly which colors are to be displayed by your browser. Color is discussed in more detail in Module 4.

Notice that the background attribute is different from the src attribute. It only lists the name of the file that is used for the background on your page, because the image is in the same folder as the HTML file itself:
Now, look at the SRC code:

```html
<p><img src="images/simpleimage.gif" width="299" height="122"> </p>
```

Notice that “images/” comes before the filename. In this case, your image is in a different folder (named images) than the source code, and you have to give the browser more specific information on where to find it. As you’ll see in the next module, it’s very important to plan the layout of your site so that you can easily find and file the pictures and other resources you want to use. Otherwise, you might end up with the dreaded (and unprofessional) broken image link symbol on your otherwise beautiful web page.

The last attribute to discuss is the `<a href>` combination that you see near the bottom of the sample code. This is the place where you can insert a link into your page. Again, notice how it comes before the words it modifies (Visit Osborne Publishing) and that it is very specific in nature. While it’s getting increasingly common for users to just type a few words into the location bar of their browser and zoom off to a website, when you insert a hyperlink, you need all the information included, including the information that specifies you want your files transported using a process known as Hypertext Transfer Protocol (http://)—the web protocol for linking one page with another.

**Hint**

The hyperlink in our sample is called an *absolute* link because it is outside the website you’re currently in. Absolute links always require the entire URL. *Relative* links, or those within your own site, can be linked by filename, as you’ll see in Module 5.

One other type of tag that hasn’t been discussed yet is known as a *metatag*—a specialized tag that is located within the head of pages and that provides hidden information about the page, its author, and its contents. You’ll see when working with Dreamweaver that this is one of the ways that you can be sure you get visitors to your web pages, because the information inside your metatags is often what search engines actually search for. Right now, you just need to know that HTML has that capability.

And, of course, HTML has many other capabilities as well, but for right now, understanding how tags and attributes work in combination with each other is all you actually need to know about HTML. Table 1-2 lists the most common tags that you’re likely to see when working with Dreamweaver.
Module 1: Fundamentals of the World Wide Web

**Tag**
- `<a href="URL">`: Creates a hyperlink to the specified URL
- `<b>`: Bolds text
- `<blockquote>`: Indents text left and right
- `<body>`: Defines the visible portion of the web page
- `<br>`: Creates a line break
- `<font>`: Describes the font (text style) to be used
- `<form>`: Creates a form
- `<frame>`: Defines a single part of a framed web page
- `<frameset>`: Defines a set of frames in a web page
- `<h1>`: The largest headline-styled text
- `<h6>`: The smallest headline-styled text
- `<head>`: Defines information and instructions that do not appear on the page itself
- `<html>`: Defines the file as an HTML document
- `<hr>`: Inserts a horizontal line (rule)
- `<i>`: Creates italic text
- `<img>`: Inserts an image
- `<ol>`: Creates a numbered (ordered) list
- `<p>`: Formats a block of text as a separate paragraph
- `<table>`: Creates a table
- `<td>`: Defines table divisions—separate cells in a table
- `<th>`: Creates a header for a table
- `<title>`: Describes the text to be displayed in the browser title bar
- `<tr>`: Defines table rows

**Attributes**
- `<body background="imagename">`: Sets the image to be used as a page background
- `<body bgcolor=?>`: Sets the page background color
- `<body text=?>`: Sets text color in the body of a page
- `<div align=?>`: Used to format blocks of text
- `<font color=?>`: Sets the color of text
- `<font size=?>`: Sets the size of text
- `<table border=?>`: Sets the width of the border around a table
- `<table width=# or %>`: Sets the width of a table as an absolute number of pixels or as a percentage of the page
- `<tr align=?>`: Sets alignment of individual cells

**Table 1-2** Basic HTML Tags and Attributes
1-Minute Drill

- Describe the function of HTML tags.
- Describe the function of HTML attributes.
- What two items of information must be included when inserting an image in a web page?

The Present and Future of HTML

The development of the Internet has happened in an amazingly short period of time. Along with this rapid development, an incredibly competitive marketplace has developed in which those who develop the latest and greatest Internet capabilities for their browsers can quickly overtake and dominate their competitors. The problem is that one browser may not display code written for another flavor of browser, and visa versa. HTML standards quickly became an issue of great concern to developers worldwide as growing differences in browser technologies indicated that entire web sites soon would need to be developed for each browser platform.

The World Wide Web Consortium (W3C, at www.w3c.org) was formed as a means of addressing these growing problems. The W3C is the organization responsible for ensuring that a common set of HTML standards is developed and maintained, and that the capabilities of those standards are widely available to webmasters.

So, with a common set of standards, developers should have no problems at all, right? Unfortunately, that isn’t the case.

Remember that the work of displaying your pages is done at the viewer’s computer and by the browser they have installed. Currently, all browsers handle the HTML standards developed by the W3C through version 2—but, as of this writing, the W3C has developed standards through version 4.01! Furthermore, neither Netscape Navigator nor Internet Explorer, even in their latest versions, supports all the tags in HTML 4.01, or they interpret the code differently. This can cause web authors and developers real headaches as they try to construct pages that will display properly no matter what type of browser is being used. As you’ll see in Module 2, the result of all this is that you need to have a very good understanding of who will be viewing your pages, what kind of browser they are likely to have, and how this impacts the features you want to make available through your code.
Module 1: Fundamentals of the World Wide Web

As a sophisticated web authoring tool, Dreamweaver 4 includes ways to address browser compatibility. These compatibility problems are discussed throughout this book, and you'll quickly come to appreciate the variety of ways in which Dreamweaver enables you to test your project.

**Hint**
To check for browser compatibility, you need to install at least the two most common browser programs—Netscape Navigator and Internet Explorer. They can be downloaded at www.netscape.com and www.microsoft.com, respectively.

What You See Is What You Get

After you understand the basics of what HTML does and how it interacts with your browser, it won't be nearly as intimidating. In fact, in the early days of the Internet, almost all web pages were created by hand—that is, raw code was written in its HTML text form. Many of these early pioneers taught themselves how to do their coding by carefully studying the code they saw on web pages and emulating (okay, sometimes copying) it. To this day, plenty of hard-core HTML junkies won't build their web pages any other way.

For the rest of us, those who see a huge page with streams of hard-to-follow commands and quickly feel overwhelmed, Dreamweaver offers the ability to design pages in a much friendlier environment, one that enables you to actually see what's going to be on your page as you work. Dreamweaver enables you to design in an almost real-time environment, and is the best software available for those who want to work in this what you see is what you get (WYSIWYG) mode. With Dreamweaver, you can apply elements to your pages, change the text size and style, insert images, and control the overall look of your pages without ever looking at HTML at all.

In addition, Dreamweaver provides clean, standards-driven, code that can easily be modified in its raw code directly within the program. Version 4 has even added the capability of viewing two simultaneous windows, displaying the design version of the page in one window, while the code itself is displayed (and automatically updated) in its pure form, as shown in Figure 1-4. Dreamweaver calls this “round-trip” HTML because page authors can switch seamlessly between a code view and a design view without worrying about the program creating code that is not compatible with current standards. This may not seem important to you right now, but at some point in the process of becoming an accomplished web author and site developer, you're going to want to have access to options that are more easily created and modified in “raw” HTML.
For the new author, or even someone who has previously done their entire HTML coding by hand, Dreamweaver also provides some other useful coding help:

- Automatically corrects overlapping and redundant tags and attributes
- Highlights incorrect code and code that is not supported by current standards
- Includes a comprehensive reference for HTML, JavaScript, and cascading style sheets (CSS)

### 1-Minute Drill

- What organization maintains and establishes standards for HTML?
- Describe how browser compatibility issues can impact web design.

- The World Wide Web Consortium maintains and establishes the HTML standards.
- Tags and attributes can be read and interpreted (or not read at all) by different browsers or even different versions of the same browser.
Module 1: Fundamentals of the World Wide Web

Project 1-1: Viewing Source Code

Many web authors got their start in web design by doing something very simple—they looked at the code contained in other web pages through the View command in their browser. You can also learn a great deal about HTML and how tags and attributes are employed by examining the source code of web pages.

You’ll start by looking at one of the world’s most popular sites, Yahoo! (www.yahoo.com). Yahoo! is an example of an Internet directory—specially constructed web pages filled with links to other sites listed by category. You can almost think of it as Yellow Pages for the Web. This section looks at both the page itself and the underlying code.

Step-by-Step

1. Open Yahoo! in your browser and you’ll see a very tightly organized page with a huge number of hyperlinks, shown in Figure 1-5.

2. Choose View | Page Source (Netscape) or Source (Internet Explorer), and you’ll see the HTML code for that page, as shown in Figure 1-6.

3. Now identify some of the tags you see:
   - What tag do you find most often on this page?
   - What is the page title?
   - Does the page use tables to lay out its content? What tells the browser to develop the tables?
   - Do you see any tags you don’t recognize?

Note

Yahoo! uses a number of special HTML tags called cascading style sheets. These special formatting tags will be discussed in Module 8.

4. Next, visit a few of the following web sites:
   - www.osborne.com
   - www.nbci.com
   - www.microsoft.com
   - www.apple.com
   - www.zdnet.com

On each of the sites, do the following:

a. Locate the code for an image that is on the page. What is the name of the image? Is the image in the same folder as the main page? If it’s in another folder, can you identify the name of the folder that holds it?
b. Try and locate special text attributes on the page. Do you find instances of the bold or italic tags? How about tags for text color?

c. Notice how tables are used in the construction of the page. Can you identify the attributes that define the table and its contents?

Figure 1-5  Yahoo! is one of the world’s most visited web sites

Figure 1-6  Viewing the code for the Yahoo! home page
Module 1: Fundamentals of the World Wide Web

d. Look for tags and attributes listed on these pages that haven’t been discussed yet. Can you tell what they do?

e. Look for a tag that is formatted like this: <!--some text in here-->. What do you think the purpose is of this kind of tag?

f. Look at the exact same web site using both Internet Explorer and Netscape Navigator. Do you see any difference in how the two browsers display the page you’re viewing?

What to Take Away

Understanding HTML and how web pages are created is not an overwhelming chore. Just like with any craft, designing web pages requires patience and lots of practice. If you’re willing to put the time into learning the software, and then practice your skills using the projects in this book, you’ll find that within a very short time, you will be confidently building web pages and assembling them into well-defined web sites. It’s going to be a lot of fun—and you’ll get started in the next chapter by looking at web site planning and how you use Dreamweaver to build your site from the ground up.

Mastery Check

1. What is the term to describe a web page’s address on the Internet?

2. Every web page begins and ends with what tag?

3. What is the function of this tag: <a href=“http://www.osborne.com”>?

4. What is the term used by Dreamweaver to describe its standards-based HTML code?

5. What is the difference between a relative link and an absolute link?